Lecture 7: Categories

Lecturer: Max S. New
Scribe: Han Jiang

Feb. 01, 2023

1 Definition

A *category* $\mathcal{C}$ is an algebraic structure that consists of:

- $\mathcal{C}_0$, a set of *objects*
- For each $a, b \in \mathcal{C}_0$, a set $\mathcal{C}_1(a, b)$ of *arrows* (also called *morphisms*) from a to b. For $f \in \mathcal{C}_1(a, b)$ when the category is clear from context, we write $f : a \to b$.
- For each $a \in \mathcal{C}_0$ a distinguished *identity* morphism $id_a \in \mathcal{C}_1(a, a)$.
- For each $a, b, c \in \mathcal{C}_0$, a *composition* operation $\circ : (\mathcal{C}_1(b, c) \times \mathcal{C}_1(a, b)) \to \mathcal{C}_1(a, c)$
- Composition respects the identity morphisms: for any $f : a \to b$, we have

  $$id_b \circ f = f$$

  and

  $$f \circ id_a = f$$

- Composition is associative: wherever the composition is defined, we have $f \circ (g \circ h) = (f \circ g) \circ h$.

2 Examples

We will give several examples of categories to help with understanding the definition.

2.1 Category of Sets (Set)

1. The objects of Set_0 are all small sets.
2. The morphisms are the functions, defined as $Set_1(A, B) = \{f | f : A \to B\}$.
3. The identity morphism is the identity function, defined as $id_A(x) = x$.
4. Composition operation is function composition, defined as $(f \circ g)(x) = f(g(x))$. We will now show the two properties hold. To show two functions are equal it is sufficient to show they are equal when applied to any element of their domain:
   - Composition respects identity morphism:
     $(id_A \circ f)(x) = id_A(f(x)) = f(x)$
     $(f \circ id_A)(x) = f(id_A(x)) = f(x)$
   - Composition is associative:
     $(f \circ (g \circ h))(x) = f((g \circ h)(x)) = f(g(h(x))) = (f \circ g)(h(x)) = ((f \circ g) \circ h)(x)$

The next two subsections will show two examples of constructing subcategories based on the category of sets, one way is having fewer ”objects”, and the other way is having fewer morphisms.

2.2 Category of Finite Sets ($FinSet$)

1. Objects of $FinSet_0$ are all finite sets[1].
2. Morphisms are all functions between the sets, so the same as with Set: $FinSet_1(A, B) = \{f | f : A \to B\}$.
3. All proceeding parts are the same as Set_0 since the morphisms are all functions between finite small sets, the proofs will be the same.

2.3 Category of Sets with only Injective Functions (Inj)

1. Objects of Inj_0 are all small sets.
2. Morphisms are the *injective* functions: $Inj_1(A, B) = \{f : A \to B | \forall x, y, f(x) = f(y) \Rightarrow x = y\}$
3. Since the composition of injective functions is also an injective, the proceeding parts are also the same as Set_0

[1]note that all finite sets are small

2.4 Category of Relations (Rel)

1. Objects of Rel_0 are all small sets.

2. Morphisms: $Rel_1(A, B) = \{R | R \subseteq A \times B\}$

3. The identity morphism is defined as $id_A = \{(a, a) | a \in A\}$

4. Let $r \in Rel_1(B, C)$, $s \in Rel_1(A, B)$, the composition is defined as:
   $(r \circ s) = \{(a, c) \in A \times C | \exists b \in B, s.t.(a, b) \in s \wedge (b, c) \in r\}$

5.
   - Composition respects identity morphism:
     Let $r \in Rel_1(A, B)$, we have

$$\begin{aligned} r \circ id_A &= \{(a, b) \in A \times B | \exists a' \in A.(a, a') \in id_A \wedge (a', b) \in r\} \\ &= \{(a, b) \in A \times B | (a, b) \in r\} \\ &= r \end{aligned}$$

   - Composition is associative as well due to some basic properties of existential quantifiers and conjunction:

$$\begin{aligned} r \circ (s \circ t) &= \{(a, d) \in A \times D | \exists c \in C.(\exists b \in B.(a, b) \in s \wedge (b, c) \in t) \wedge (c, d) \in r\} \\ &= \{(a, d) \in A \times D | \exists b \in B.(a, b) \in s \wedge (\exists c \in C.(b, c) \in t \wedge (c, d) \in r)\} \\ &= (r \circ s) \circ t \end{aligned}$$

We have been discussing the objects on mathematical objects, now we will switch to syntactic objects, which will come up when we try to formalize the soundness and completeness theorem, as well as the initiality theorems.

2.5 Syntactic STT ($STTerm$)

1. $STTerm_0 =$ STT types.

2. $STTerm_1(A, B) = \{M | x : A \vdash M : B\}$

3. $id_A := x : A \vdash x : A$

4. Let M be specified as part 2, and $N \in STTerm_1(B, C)$, $M \circ N := M[N/x]$

5.
   - Composition respects identity morphism:
     $M \circ id = M[x/x] = M \overset{hw2}{=} x[M/x] = id \circ M$
   - Composition is associative:
     $(M \circ N) \circ P = (M[N/x])[P/x] \overset{hw2}{=} M[N[P/x]/x] = M \circ (N \circ P)$

2.6 Simple Type Theory Lambda ($STT\lambda$)

1. $STT\lambda_0 =$ STT types.
2. $STT\lambda_1(A, B) := \{M | \cdot \vdash M : A \Rightarrow B\}$
3. $id_A = \lambda x.x$
4. Let $M : B \Rightarrow C$ and $N : A \Rightarrow B$, $M \circ N := \lambda x.M(Nx) : A \Rightarrow C$.
5. - Composition respects identity:

$$\begin{aligned} M \circ id &= \lambda x.M((\lambda y.y)x) \\ &\overset{\beta}{=} \lambda x.Mx \\ &\overset{\beta}{=} \lambda x(\lambda y.y)(Mx) \\ &= id \circ M \end{aligned}$$

   - Composition is associative:

$$\begin{aligned} M \circ (N \circ P) &= \lambda x.M((N \circ P)x) \\ &= \lambda x.M((\lambda x.N(Py))x) && (\beta) \\ &= \lambda x.M(N(Px)) && (\beta) \\ &= \lambda x.(\lambda z.M(Nz))(Px) \\ &= \lambda x.(M \circ N)(Px) \\ &= (M \circ N) \circ P \end{aligned}$$

2.7 Simple Type Theory Context ($STTCtx$)

1. $STTCtx_0 = \Gamma$ ctx, which means the objects are not types but contexts.
2. $STTCtx_1(\Delta, \Gamma) := \prod_{x:A\in\Gamma} \Delta \vdash \cdot : A$.
3. Identity and Composition as well as the related properties are left as exercises in HW3.

2.8 Category of Posets ($Poset$)

1. $Poset_0 =$ small partial ordered sets.
2. $Poset_1(P, Q) := monotonefunctionsf : P \to Q$.
3. Identity is monotone and composition of monotone functions is monotone.

2.9 Category of Graphs ($Graph$)

1. $Graph_0$ = Small directed multigraphs (with loops).
2. $Graph_1(G, H) := (F_v : G_v \to H_v) \times (F_e : G_e \to H_e)$ such that the edge functions preserve the sources and targets of the edges.

2.10 Category of Heyting Algebras ($HeytAlg$)

1. $HeytAlg_0$ = Heyting Algebras.
2. $HeytAlg_1(H_1, H_2) := f : H_1 \to H_2$ such that f is monotone and preserves all the Heyting algebra structures. (e.g. $f(x \wedge y) = f(x) \wedge f(y)$)

2.11 Category of Matrices ($Matrices$)

1. $Matrices_0 := \mathbb{N}$.
2. $Matrices_1(m, n) := m \times n$ matrix.
3. $id_n = I_n$
4. $M \circ N = MN$ (Matrix multiplication, properties follows from the restriction on multiplications).

2.12 Category of Vector Spaces ($Vect$)

1. $Vect_0 :=$ Vector Spaces
2. $Vect_1(V, M) :=$ linear transformations.
3. All proceeding parts follows from basic rules of Linear Algebra.

2.13 Category of Monoids ($Monoid$)

A monoid is a set X paired with an identity element e and a function m, details will be shown below:

1. $Monoid_0 := X \times (e : X) \times (m : X^2 \to X)$, m and e satisfies
   - (Identity) $m(e, x) = x = m(x, e)$
   - (Associativity) $m(x, m(y, z)) = m(m(x, y), z)$
2. $Monoid_1(X, Y) := f : |X| \to |Y|$ such that
   - $fe_x = e_y$
   - $f(m_x(x_1, x_2)) = m_y(fx_1, fx_2)$

3 General Families of Categories

In last class we saw that we can upgrade any preorder to category, and category is just the preorder where we care why $X \leq Y$. Therefore in general, we can define when a category is equivalent to a preorder, and that is called a *thin category.*

Definition 1. *A thin category, $\mathcal{C}$, is the category where $\forall a, b \in \mathcal{C}$, $|\mathcal{C}_1(a,b)| \leq 1$.*

Note that thin categories are the categories that are equivalent to preorders, and in those categories we mostly care about the objects, and neglect the arrows. In the opposite extreme, we have categories where we barely care about the object:
Consider a category $\mathcal{C}$, where

- $\mathcal{C}_0 = *$.
- $id_* \in \mathcal{C}_1(*, *)$.
- For $f, g \in \mathcal{C}_1(*, *)$, we have $f \circ g \in \mathcal{C}_\infty(*, *)$
- Notice that in $\mathcal{C}$ the only object is * and the only morphism is id_*, thus all the related properties are trivial.

Note that the category above is exactly the data of a monoid: the morphisms of the one-object category are the elements and the identity and composition are the identity element and multiplication of a monoid. The category axioms specialize to precisely the monoid axioms in this case.

From the two extreme cases above, we will notice that preorders and monoids can both be seen as special types of categories.

4 Basic Notions Inside Categories

We will be looking at the structures that we are familiar from math (usually sets of functions), and see if we can generalize those operations to categories and if they always behave the same way (The answer is usually no). We will first take a look at isomorphism:
In set theory, we know that two sets A and B are isomorphic if there is a bijective function $f : A \to B$. Note that bijective function is defined in two ways

- $f : A \to B$ is both injective and surjective
- f has an inverse $f^{-1} : B \to A$ such that $\forall x \in A, y \in B$, $f^{-1}(f(x)) = x$ and $f(f^{-1}(y)) = y$

We will try to generalize the notion of bijective function categories:

- Fix arbitrary category $\mathcal{C}$.
- Let $f : A \to B$

- We will look at both definitions and try to generalize.

  1. For the second definition, we cannot use parenthesis on the arrows of an arbitrary categories, but we can utilize the notion of composition and identity: $\exists f^{-1} : B \to A$ such that $f^{-1} \circ f = id_A$ and $f \circ f^{-1} = id_B$.
  2. For the first definition, we need to formally define injectivity and surjectivity in an arbitrary category. Consider injectivity, the definition on set theory is $\forall x, y \in A, f(x) = f(y) \Rightarrow x = y$. However, in an arbitrary category we do not even know that if the object of $\mathcal{C}$ is a structure that can contain elements (for example, types or *). Therefore, we need to generalize the notion of "having an element" for an object in a category. Notice we can view a set A having an element x as having a morphism from the one element set 1 to A: $1 \xrightarrow{x} A$. For the general category, we can phrase the morphism as $\forall C, C \to A$. Therefore we can rephrase the notion of injectivity as

  $$\forall C, \forall x, y \in \mathcal{C}_1(C, A), f \circ x = f \circ y \Rightarrow x = y \tag{1}$$

     We call this motion a *monomorphism* (or *monic*)

  What about surjectivity? Here we have to be careful because there are multiple equivalent ways of defining a surjective function that are not equivalent when we generalize. Here are two of them:

  – A morphism $f : A \to B$ is an *epimorphism* if $\forall k_1, k_2 \in \mathcal{C}_a(B, C)$ if $k_1 \circ f = k_2 \circ f$ then $k_1 = k_2$.

  – A morphism $f : A \to B$ is a *retraction* or a *split epimorphism* if for every generalized element $b : X \to B$, there exists a generalized element $a : X \to A$ such that $f \circ a = b$.

    Note that by picking b to be $\mathrm{id}_B : B \to B$, this gives us a morphism $s : B \to A$ satisfying $f \circ s = \mathrm{id}_B$, so this is precisely saying that f has a "pre-inverse" (though not necessarily a post-inverse).

  The latter definition in terms of generalized elements is clearly a generalization of the familiar definition of a surjective function. The former is less familiar, but it is easy to see that every split epimorphism is an epimorphism (exercise). In the category of sets in ZFC, every epimorphism is a split epi. In fact, the statement that every epimorphism in the category of sets is a retraction is *precisely* the axiom of choice.

  Then can we reproduce the classical theorem that every injective and surjective function has an inverse?

  – If we replace injective by monomorphism and surjective by split epimorphism then it is true (exercise).

- But if we replace surjective by only epimorphism, then it fails in the category of monoids: the monoid homomorphism that is the inclusion of the additive monoid of natural numbers into the additive monoid of integers is mono and epi (exercise), but clearly not an isomorphism.